Lecture 19 Quiz

David Kirby

Due: 07 April 2020

1. **Define a Harvard architecture and a von Neumann architecture.**

   Harvard architecture stores instructions and data in separate memories while von Neumann architecture stores instructions and data in the same memory.

2. **In our hazard detection logic, how exactly do we account for instructions that do not use the `RS` and `RT` field?**

   If `UseShamt` is asserted then `RS` is not being used, if `UseImmediate` is asserted then we are in fact not using `RT` and we have to add in the logic to account for this to prevent unnecessary slips of the pipeline. We do this by using if statements to check that `UseShamt` or `UseImmediate` are not equal to zero.

3. **Why is the addi not a good candidate for being at the branch delay slot?**

   The compiler must make certain that whatever is placed in the branch delay slot is not going to cause any harm if the branch is not taken, such as a segmentation fault.

4. **Define branch aliasing.**

   In our basic 1-bit wide branch history table, we typically have close to 2^{10} or 1024 entries in the table. This means it's possible that two branches map to the same location in the table, so they fight each other; however, this is a prediction and doesn't necessarily have to be correct. The more correct it is, the higher performance we get; the less correct it is, the lower performance.